Local Generic Position for Root Isolation of Zero-dimensional Triangular Polynomial Systems

Jia Li, Jinsan Cheng, Elias Tsigaridas

Local Generic Position for Root Isolation of Zero-dimensional Triangular Polynomial Systems

Jia Li[1], Jin-San Cheng[2], and Elias P. Tsigaridas[3]

[1] Beijing Electronic Science and Technology Institute lijia@besti.edu.cn
[2] KLMM, AMSS, Chinese Academy of Sciences jcheng@amss.ac.cn
[3] POLSYS project, INRIA, LIP6/CNRS elias@polsys.lip6.fr

Abstract. We present an algorithm based on local generic position (LGP) to isolate the complex or real roots and their multiplicities of a zero-dimensional triangular polynomial system. The Boolean complexity of the algorithm for computing the real roots is single exponential: $\tilde{\mathcal{O}}_B(N^{n^2})$, where $N = \max\{d, \tau\}$, d and τ, is the degree and the maximum coefficient bitsize of the polynomials, respectively, and n is the number of variables.

1 Introduction

Solving polynomial systems is a basic problem in the fields of computational sciences, engineering, etc. Usually, the polynomial systems are transformed into triangular polynomial systems by algebraic methods, such as Gröbner bases, characteristic sets, CAD, and resultants. In most case, we consider zero-dimensional polynomial systems. So in the end, we need to solve zero-dimensional triangular systems. One practical problem is to determine the topology of real algebraic curves or surfaces with CAD based method [4, 8, 17], we need to isolate the real roots of a zero-dimensional triangular system with multiple zeros. We will discuss how to solve this kind of system in this paper.

A zero-dimensional triangular system has the form $\Sigma_n = \{f_1, \ldots, f_n\}$, where $f_i \in \mathbb{Q}[x_1, \ldots, x_i]$ $(i = 1, \ldots, n)$, $\mathbb{Q}$ is the field of rational numbers. Our aim is to find zeros $\boldsymbol{\xi}^n = (\xi_1, \ldots, \xi_n) \in \mathbb{C}^n($ or $\mathbb{R}^n)$ of Σ_n, where $\mathbb{C}, \mathbb{R}$ are the field of complex and real numbers, respectively.

A local generic position method (shortly LGP) was introduced in [6]. The method was used to solve bivariate polynomial systems and the experiments show that the method works well. The method was extended to solve general zero-dimensional system by computing Gröbner basis at first and then computing a linear univariate representation [7]. In this paper, we will extend the LGP method to solve general zero-dimensional triangular system by computing resultant only. The complexity analysis and the experiments show the effectivities and efficiency of the algorithm.

For the system $\Sigma_{i+1} (i \geq 1)$, we can assume that we have got the zeros of Σ_i. For any fixed zeros of Σ_{i-1} (it may be $\{0\}$), (f_i, f_{i+1}) can be regarded as a bivariate polynomial system. We shear the hypersurfaces (surface, curve) defined by the polynomials f_i, f_{i+1} on a special direction such that the first $i-1$ coordinates and the $(i+1)$-th coordinate are unchanged. The new system is denoted as Σ'_{i+1}. Then we project all the zeros of Σ'_{i+1} to the i-dimensional space by eliminating the $(i+1)$-th coordinate, denoted as Σ^*_i. Solving Σ^*_i, we can recover the roots of Σ_{i+1} by the LGP method with the zeros of Σ_i, Σ^*_i. Step by step, we can get all the zeros of the system. And the method keeps the multiplicity of each zero of the given system. In the end, we get an algebraic representation for the zeros of the system Σ_n: each coordinate of each zero is a linear combination of some zeros of several univariate polynomials. From the algebraic representation, we can get the zeros of the system under any given precision.

The bit complexity of our algorithm for real roots is $\tilde{\mathcal{O}}_B(N^{n^2})$, where N, n will be defined in Section 4. Our method is ***complete*** in the sense that Σ_n can be any zero-dimensional triangular system.

Root-isolating of zero-dimensional triangular system is studied before. Most of the methods can not deal with triangular system with multiple zeros directly [10, 12, 19, 5, 23]. Usually, they decompose the system into triangular systems without multiple zeros and then isolate the real zeros of them. Cheng et al [9] provides a direct method, although their method does not give an algebraic representation of the real zeros and can not give the multiplicities of the zeros. In [26], they provide a method to compute the multiplicities of the real zeros when they compute the zero by existing methods. There are some related work about algebraic representation. Gao and Chou [16] privide a method to represent the zeros of a radical characteristic set. From a Gröbner basis, a rational representation of the zeros of a system is provided and the representation depends on the multiplicities of the solutions [1]. Fouillier [21] uses rational univeriate representation to represent the zeros of a polynomial system by computing the Gröbner basis of the system.

2 Zero-dimensional triangular system solving

In this section, we give the basic theory for our method.

Let $\Sigma_i = \{f_1(x_1), f_2(x_1, x_2), \ldots, f_i(x_1, x_2, \ldots, x_i)\} \in \mathbb{Q}[x_1, x_2, \ldots, x_i] (i = 1, \ldots, n)$ be a general zero-dimensional triangular system. $\boldsymbol{\xi}^i = (\xi_1, \ldots, \xi_i) \in \mathrm{Zero}(\Sigma_i)$, where $\mathrm{Zero}(t)$ represents the zero set of $t = 0$. And t can be a polynomial or a polynomial system.

Let $f \in \mathbb{C}[x]$. Then the separation bound $\mathrm{sep}(f)$ and root bound $\mathrm{rb}(f)$ of f are defined as follows: $\mathrm{sep}(f) := \min\{\Delta(\alpha, \beta) | \forall \alpha, \beta \in \mathbb{C}\, s.t. f(\alpha) = f(\beta) = 0, \alpha \neq \beta\}$, where $\Delta(\alpha, \beta) := \min\{|\mathrm{Re}(\alpha - \beta)|, |\mathrm{Im}(\alpha - \beta)|\}$, $\mathrm{Re}(\alpha - \beta), \mathrm{Im}(\alpha - \beta)$ are the real part and imaginary part of $\alpha - \beta$ respectively. We also need the definition of the root bound: $\mathrm{rb}(f) := \max\{|\alpha| | \forall \alpha \in \mathbb{C}\, s.t. f(\alpha) = 0\}$.

Assume that we have solved the system $\Sigma_i (1 \leq i \leq n-1)$. The assumption is reasonable since we can solve Σ_1 directly with many existing tools, such as [22, 25]. And we can get a separation bound r_1 of the roots of $f_1(x_1) = 0$. Based on the roots of $f_1 = 0$, we can estimate the root bound R_2.

Let $r_j (1 \le j \le i)$ be a positive rational number, such that

$$r_j \le \frac{1}{2} \min_{\boldsymbol{\xi}^{j-1} \in \mathrm{Zero}(\Sigma_{j-1})} \mathrm{sep}(f_j(\boldsymbol{\xi}^{j-1}, x_j)). \tag{1}$$

We can compute r_j after we get the roots of $f_j(\boldsymbol{\xi}^{j-1}, x_j) = 0$.

Based on the zeros of Σ_j, we can estimate the root bound on x_{j+1} (we will show how to estimate the bound later) to get a positive rational number R_{j+1}, such that

$$R_{j+1} \ge \max_{\boldsymbol{\xi}^{j} \in \mathrm{Zero}(\Sigma_{j})} \mathrm{rb}(f_{j+1}(\boldsymbol{\xi}^{j}, x_{j+1})). \tag{2}$$

We usually add a previously estimated value, say r'_{j+1}, for r_{j+1} to the above root bound to ensure that after shearing and projection, the fixed neighborhoods of the zeros of $T_i^i(X_i^i)$ (see definition below) are disjoint. Then when we compute r_{j+1}, we choose the one no larger than r'_{j+1}.

We say two plane curves defined by $f, g \in \mathbb{C}[x, y]$ s.t. $\gcd(f, g) = 1$ are in a **generic position** w.r.t. y if (1) The leading coefficients of f and g w.r.t. y have no common factors, and (2) If h is the resultant of f and g w.r.t. y, then any $\alpha \in \mathbb{C}$ such that $h(\alpha) = 0$, $f(\alpha, y), g(\alpha, y)$ have only one common zero in $\mathbb{C}$.

Now we introduce *local generic position* [6, 7]. Given $f, g \in \mathbb{Q}[x, y]$, not necessarily in generic position, we considerthe the mapping $\phi : (x, y) \to (x + s\,y, y), s \in \mathbb{Q}$, with the following properties: (i) $\phi(f), \phi(g)$ are in a generic position w.r.t. y, and (ii) Let $\bar{h}$, h be the resultants of $\phi(f), \phi(g)$ and f, g w.r.t. y, respectively. Each root α of $h(x) = 0$ has a neighbor interval H_α such that $H_\alpha \cap H_\beta = \emptyset$ for roots $\beta \ne \alpha$ of $h = 0$. And any root (γ, η) of $f = g = 0$ which has a same x-coordinate γ, is mapped to $\gamma' = \gamma + s\,\eta \in H_\gamma$, where $h(\gamma) = 0, \bar{h}(\gamma') = 0$, as shown in Figure 1. Thus we can recover $\eta = \frac{\gamma' - \gamma}{s}$.

2.1 Basic theory and method

For each $\boldsymbol{\xi}^i = (\xi_1, \ldots, \xi_i) \in \mathrm{Zero}(\Sigma_i)$, the roots of $f_{i+1}(\boldsymbol{\xi}^i, x_{i+1}) = 0$ are bounded by R_{i+1}. We can take a shear mapping on $f_{i+1}(x_1, \ldots, x_{i+1})$ such that when projected to i-D space, all the roots of $f_{i+1}(\boldsymbol{\xi}^i, x_{i+1}) = 0$ are projected into the fixed neighborhood of ξ_i (centered at ξ_i bounded by $r_i/2$). This can be achieved by take the following shear mapping on (x_i, x_{i+1}).

$$X_2^{i+1} = x_i + \frac{r_i}{R_{i+1}} x_{i+1},\ X_1^{i+1} = x_{i+1}. \tag{3}$$

Applying (3) to the system Σ_{i+1}, we derive a new system $\Sigma'_{i+1} = \{f_1(x_1), \ldots, f_{i-1}(x_1, \ldots, x_{i-1}), f_i(x_1, \ldots, x_{i-1}, X_2 - \frac{r_i}{R_{i+1}} X_1^{i+1}), f_{i+1}(x_1, \ldots, x_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1}, X_1^{i+1})\}$. There is only one root of $f_{i+1}(\xi_1, \ldots, \xi_{i-1}, \theta_2 - \frac{r_i}{R_{i+1}} X_1^{i+1}, X_1^{i+1}) = 0$ corresponding to each i-D root $(\xi_1, \ldots, \xi_{i-1}, \theta_2) \in \mathrm{Zero}(\Sigma_i^*)$. As is shown in Figure 1, θ_2 is some dot point on x_i-axis, corresponding to each dot point, there is only one triangle point. Let

$$T_2^{i+1}(x_1, \ldots, x_{i-1}, X_2^{i+1}) = \mathrm{Res}_{X_1^{i+1}}(f_i(x_1, \ldots, x_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1}), f_{i+1}(x_1, \ldots, x_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1}, X_1^{i+1})),$$

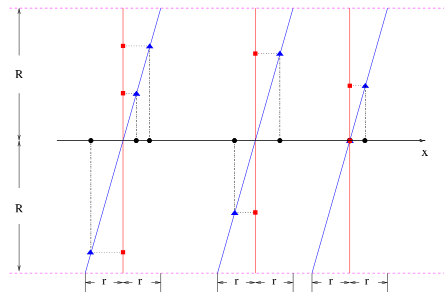

Fig. 1. Local generic position

where $\mathrm{Res}_t(f, g)$ is the resultant of f and g w.r.t. t. Then we get a triangular system $\Sigma_{i-1} \cap \{T_2^{i+1}\}$. We will further study the relationship between the zeros of Σ_{i+1} and $\Sigma_{i-1} \cap \{T_2^{i+1}\}$ below. Considering the multiplicities of the zeros, we give the following lemma.

Lemma 1. *For each zero $\boldsymbol{\xi}^i$ of Σ_{i-1}, there exists a one to one correspondence between the roots of $\{f_i(\xi_1, \ldots, \xi_{i-1}, x_i), f_{i+1}(\xi_1, \ldots, \xi_{i-1}, x_i, x_{i+1})\} = 0$ and the roots of $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$, and the multiplicities of corresponding zeros in their equation(s) are the same.*

Lemma 2. *There exists a one to one correspondence between the zeros of triangular systems Σ_{i+1} and $\Sigma_{i-1} \cap \{T_2^{i+1}(x_1, \ldots, x_{i-1}, X_2^{i+1})\}$. And the corresponding zeros have the same multiplicities in their system.*

Proof. Since both the systems have a same sub-system Σ_{i-1}, we can derive that the lemma is correct by Lemma 1.

Lemma 3. *For $(\xi_1, \ldots, \xi_i) \in \mathrm{Zero}(\Sigma_i)$, the roots of $f_{i+1}(\xi_1, \ldots, \xi_i, x_{i+1})$ are:*

$$x_{i+1} = \frac{R_{i+1}}{r_i}(\zeta_2 - \xi_i), T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, \zeta_2) = 0 \ \textit{and} \ |\zeta_2 - \xi_i| < r_i. \tag{4}$$

Proof. The first formula is directly derived from (3). Note that the first formula just holds for ζ_2' corresponding zeros having ξ_i as coordinate. So the inequality holds.

The above lemma tells us how to derive the roots of $f_{i+1}(\xi_1, \ldots, \xi_i, x_{i+1}) = 0$ from the roots of $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$. From (1) and (2), the corollary below is obvious.

Corollary 1. *All the roots of $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$ are inside the fixed neighborhood of 0 (centered at 0 bounded by R_i) for all $(\xi_1, \ldots, \xi_{i-1}) \in \mathrm{Zero}(\Sigma_{i-1})$.*

We apply the previous procedure on the triangular system $\Sigma_{i-1} \cap \{T_2^{i+1}\}$ with the mapping

$$X_3^{i+1} = x_{i-1} + \frac{r_{i-1}}{R_i} X_2^{i+1}, \ X_2^{i+1} = X_2^{i+1}, \tag{5}$$

we can derive

$$T_3^{i+1}(x_1,\ldots,x_{i-2},X_3^{i+1}) = \mathrm{Res}_{X_2^{i+1}}(f_{i-2}(x_1,\ldots,x_{i-3},X_3^{i+1}-\frac{r_{i-1}}{R_i}X_2^{i+1}), T_2^{i+1}(x_1,\ldots,x_{i-3},X_3^{i+1}-\frac{r_{i-1}}{R_i}X_2^{i+1},X_2^{i+1})).$$

So, we have a triangular system $\Sigma_{i-2}\cap\{T_3^{i+1}\}$. Since Corollary 1 holds, the results in Lemma 2 still hold on $\Sigma_{i-1}\cap\{T_2^{i+1}\}$ and $\Sigma_{i-2}\cap\{T_3^{i+1}\}$. By (5), and similarly as (4), we derive

$$\zeta_2 = \frac{R_i}{r_{i-1}}(\zeta_3-\xi_{i-1}),\ |\zeta_3-\xi_{i-1}|<r_{i-1}, \quad T_3^{i+1}(\xi_1,\ldots,\xi_{i-2},\zeta_3)=0. \tag{6}$$

Then we have $x_{i+1}=\frac{R_{i+1}}{r_i}(\frac{R_i}{r_{i-1}}(\zeta_3-\xi_{i-1})-\xi_i)$, where $|\zeta_3-\xi_{i-1}|<r_{i-1}$, $|\frac{R_i}{r_{i-1}}(\zeta_3-\xi_{i-1})-\xi_i|<r_i$, $T_3^{i+1}(\xi_1,\ldots,\xi_{i-2},\zeta_3)=0$.

The above formula means that we can get the roots of $f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1})=0$ by solving $T_3^{i+1}(\xi_1,\ldots,\xi_{i-2},X_3^{i+1})=0$ directly.

Step by step, we can derive a univariate polynomial $T_{i+1}^{i+1}(X_{i+1}^{i+1})$. It holds $\zeta_i=\frac{R_2}{r_1}(\zeta_{i+1}-\xi_1)$ and $|\zeta_{i+1}-\xi_1|<r_1$. Now we can represent $\mathrm{Zero}(f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1}))$ by $\xi_1,\ldots,\xi_i$ and the roots of $T_{i+1}^{i+1}(X_{i+1}^{i+1})$, where $(\xi_1,\ldots,\xi_i)\in\mathrm{Zero}(\Sigma_i)$.

Lemma 4. *For any zero $(\xi_1,\ldots,\xi_i)\in\mathrm{Zero}(\Sigma_i)$, each root ξ_{i+1} of $f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1})=0$ is mapped to a root of $T_{i+1}^{i+1}(X_{i+1}^{i+1})=0$. And we can derive ξ_{i+1} by $T_{i+1}^{i+1}(X_{i+1}^{i+1})=0$ as follows.*

$$\begin{gathered}\xi_{i+1}=\frac{R_{i+1}}{r_i}(\zeta_2-\xi_i),\quad \zeta_2=\frac{R_i}{r_{i-1}}(\zeta_3-\xi_{i-1}),\\ \ldots,\\ \zeta_i=\frac{R_2}{r_1}(\zeta_{i+1}-\xi_1),\ T_{i+1}^{i+1}(X_{i+1}^{i+1})=0,\end{gathered} \tag{7}$$

where $|\zeta_2-\xi_i|<r_i$, $|\zeta_3-\xi_{i-1}|<r_{i-1},\ldots,$ $|\zeta_{i+1}-\xi_1|<r_1$.

Proof. Using Lemma 3 recursively, we can derive the above formula.

Lemma 5. *For any $(\xi_1,\ldots,\xi_i)\in\mathrm{Zero}(\Sigma_i)$, each distinct root ξ_{i+1} of $f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1})=0$ is mapped to a root of $T_{i+1}^{i+1}(X_{i+1}^{i+1})=0$. And we can derive ξ_{i+1} as follows.*

$$\xi_{i+1}=(\prod_{j=1}^{i}\frac{R_{j+1}}{r_j})(\eta_{i+1}-\eta_i), \tag{8}$$

where $\eta_{i+1}\in\mathrm{Zero}(T_{i+1}^{i+1})$, $\eta_i\in\mathrm{Zero}(T_i^i)$, and $|\eta_{i+1}-\eta_i|<(\prod_{j=1}^{i-1}\frac{r_j}{R_{j+1}})r_i$.

Lemma 6. *The multiplicity of the zero $(\xi_1,\ldots,\xi_i,\xi_{i+1})$ of Σ_{i+1} is equal to the multiplicity of the corresponding root in $T_{i+1}^{i+1}(X_{i+1}^{i+1})=0$.*

Proof. Using Lemma 2 recursively, we can derive the lemma.

Theorem 1. *Wih the notations above, we have the following representation for a general zero-dimensional triangular system* Σ_n*:* $\{\{T_1^1,\ldots,T_n^n\},\{r_1,\ldots,r_{n-1}\},\{R_2,\ldots,R_n\}\}$*, such that the zeros of* Σ_n *can be derived as follows.*

$$\begin{aligned}
\xi_1 &= \eta_1, \eta_1 \in \mathrm{Zero}(T_1^1),\\
\xi_2 &= \tfrac{R_2}{r_1}(\eta_2-\eta_1), \eta_2 \in \mathrm{Zero}(T_2^2), |\eta_2-\eta_1| < r_1,\\
&\cdots\\
\xi_i &= (\textstyle\prod_{j=1}^{i-1}\tfrac{R_{j+1}}{r_j})(\eta_i-\eta_{i-1}), \eta_i \in \mathrm{Zero}(T_i^i),\\
&\quad |\eta_{i+1}-\eta_i| < (\textstyle\prod_{j=1}^{i-1}\tfrac{r_j}{R_{j+1}})r_i,\\
&\cdots\\
\xi_n &= (\textstyle\prod_{j=1}^{n-1}\tfrac{R_{j+1}}{r_j})(\eta_n-\eta_{n-1}), \eta_n \in \mathrm{Zero}(T_n^n),\\
&\quad |\eta_n-\eta_{n-1}| < (\textstyle\prod_{j=1}^{n-2}\tfrac{r_j}{R_{j+1}})r_{n-1},
\end{aligned}$$

where $T_j^j(j=1,\ldots,n)$ *are univariate polynomials,* $T_1^1 = f_1$*. For each zero* $(\xi_1,\ldots,\xi_i)$ $(1 \le i \le n)$ *of the system* Σ_i*, the multiplicity of the zero in the system is the multiplicity of the corresponding zero* η_i *in the univariate polynomial* T_i^i*.*

Remark: From the second part of the theorem, we can compute the multiplicity of $\xi_{i+1} \in \mathrm{Zero}(f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1}))$ in $f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1}) = 0$, it is the multiplicity of the zero $(\xi_1,\ldots,\xi_{i+1})$ in Σ_{i+1} dividing the multiplicity of the zero $(\xi_1,\ldots,\xi_i)$ in Σ_i. It gives a simple proof for the main result in [23].

2.2 Estimation of bounds r_i, R_{i+1}

To estimate the bounds r_i, R_{i+1}, we can directly derive the bound by the method in [14]. But the derived bounds r_i is tiny and R_{i+1} is huge. We prefer to use direct methods to get the bounds.

For r_i, we can directly compute the bound on the zeros of Σ_i using (1). Let

$$S(x_{i+1}) = \mathrm{Res}_{x_1}(\mathrm{Res}_{x_2}(\cdots\mathrm{Res}_{x_i}(f_{i+1},f_i),\cdots,f_2),f_1). \tag{9}$$

Then we can estimate R_{i+1} by estimating the root bound of $S(x_{i+1})$.

The methods to estimate the bound for r_i, R_{i+1} can be used both for complex and real roots isolation. We focus on real roots isolation in this paper. So for r_i, we compute it after we get the real roots of $\Sigma_i = 0$ with the following formula. $\mathrm{sep}(f) := \min\{|\alpha-\beta| |\forall\alpha,\beta \in \mathbb{R}\, s.t. f(\alpha) = f(\beta) = 0, \alpha \ne \beta\}$.

For R_{i+1}, we at first estimate the root bound on $f_{i+1}(\xi_1,\ldots,\xi_i,x_{i+1}) = 0$ for a fixed zero $(\xi_1,\ldots,\xi_i)$. Doing so, we need to use the definition of sleeve (see [9, 18, 19] for details). Given $g \in \mathbb{Q}[x_1,\ldots,x_n]$, we decompose it uniquely as $g = g^+ - g^-$, where $g^+, g^- \in \mathbb{Q}[x_1,\ldots,x_n]$ each has only positive coefficients and with minimal number of monomials. Given an isolating box $\square\boldsymbol{\xi}^i = [a_1,b_1]\times\cdots\times[a_i,b_i]$ for $\boldsymbol{\xi}^i = (\xi_1,\ldots,\xi_i)$, we assume that $a_j, b_j, \xi_j \ge 0, 1 \le j \le i$ since we can take a coordinate system transformation to satisfy the condition when $\xi_j < 0$. Then we define

$$f^u(x) = f^u_{i+1}(\square\boldsymbol{\xi}^i; x) = f^+_{i+1}(\boldsymbol{b}_i, x) - f^-_{i+1}(\boldsymbol{a}_i, x),$$

$$f^d(x) = f_{i+1}^d(\square\boldsymbol{\xi}^i; x) = f_{i+1}^+(\boldsymbol{a}_i, x) - f_{i+1}^-(\boldsymbol{b}_i, x), \tag{10}$$

where $\boldsymbol{a}_i = (a_1, \ldots, a_i)$, $\boldsymbol{b}_i = (b_1, \ldots, b_i)$. Then (f^u, f^d) is a ***sleeve*** of $f_{i+1}(\boldsymbol{\xi}^i, x_{i+1})$. When considering $x \geq 0$, we have (see [9]):

$$f^d(x) \leq f_{i+1}(\boldsymbol{\xi}^i, x) \leq f^u(x). \tag{11}$$

If the leading coefficients of f_u and f_d have the same signs, then we can find that the root bound of $f_{i+1}(\boldsymbol{\xi}^i, x)$ is bounded by the root bounds of f_u and f_d.

Lemma 7. *[24] Let a polynomial of degree d be $f(x) = a_d x^d + a_{d-1}x^{d-1} + \ldots + a_0 \in \mathbb{R}[x], a_d \neq 0$. Let $R = 1 + \max_{0 \leq k \leq d-1} |\frac{a_k}{a_d}|$, then all zeros of $f(x)$ lie inside the circle of radius R about the origin.*

If the considered triangular system is not regular, the leading coefficients of f_u and f_d always have different signs. But the absolute value of the leading coefficients are very close to zero. So usually, the root bound of $f_{i+1}(\boldsymbol{\xi}^i, x)$ is also bounded by the larger of the root bound of f_u and f_d. Then we can get R_{i+1} by the lemma above.

The ways to compute r_i, R_{i+1} for real case usually work for our method since a random shear mapping usually puts the system into a generic position and the real roots are in a local generic position.

2.3 Precision control

When we compute the approximating zeros of a given zero-dimensional triangular system with the method we provided, the errors of the zeros will cumulate. So we need to control the error under a wanted precision. This is what we want to discuss in this subsection.

Consider the coordinate ξ_i of the zero $\boldsymbol{\xi^n} = (\xi_1, \ldots, \xi_n)$ of the triangular system Σ_n in Theorem 1. Assume that we derive the coordinate ξ_j under the precision $\rho_j(> 0)$, and we isolate the roots of $T_j^j(X_j^j) = 0$ under the precision $\epsilon_j(> 0)$, Note that $\rho_1 = \epsilon_1$.

From (8), the following lemma is clear.

Lemma 8. *With the symbol above, we can derive that the root precision ρ_i for ξ_i is defined as follows.*

$$\rho_i = (\textstyle\prod_{j=1}^{i-1} \frac{R_{j+1}}{r_j})(\epsilon_i + \epsilon_{i-1}). \tag{12}$$

From Lemma 8, we can compute the zeros of Σ_n under any given precision by controlling the precisions $\epsilon_i(1 \leq i \leq n)$. For example, we can set them as follows if we require the precision of the output zeros to be ϵ.

$$\epsilon_i = \Pi_{j=1}^{i} \frac{r_j}{R_{j+1}} \frac{\epsilon}{2} (1 \leq i \leq n-1), \epsilon_n = \Pi_{j=1}^{n-1} \frac{r_j}{R_{j+1}} \frac{\epsilon}{2}. \tag{13}$$

In order to practically avoid refining the roots when we want to control the precision under a given ϵ, we can previously assume $\frac{R_{i+1}}{r_i}$ less than a number, such as $10, 2^3$, before we solve the system. This help us to previously estimate the precisions that should be used to get the roots of $T_i^i(X_i^i) = 0 (1 \le i \le n)$.

For root isolation, we require not only the roots satisfying the given precision, but the isolating boxes being disjoint for distinct roots. We will show how to ensure that the isolating boxes are disjoint.

For real numbers α and β, $\alpha < \beta$ in $\mathbb{R}$, if we use intervals $[a, b]$ and $[c, d]$ to represent them respectively. Denote

$$|\alpha| = |b - a|, \mathrm{Dis}(\alpha, \beta) = \begin{cases} c - b, & b < c, \\ 0, & b \ge c. \end{cases}$$

For real points $\xi = (\xi_1, \ldots, \xi_n)$ and $\eta = (\eta_1, \ldots, \eta_n)$ in $\mathbb{R}^n$, if we use boxes $[a_1, b_1] \times \ldots \times [a_n, b_n]$ and $[c_1, d_1] \times \ldots \times [c_n, d_n]$ to represent them respectively. Denote

$$|\xi| = \max_{i=1,\ldots,n} \{b_i - a_i\}, \mathrm{Dis}(\xi, \eta) = \min_{i=1,\ldots,n} \{\mathrm{Dis}(\xi_i, \eta_i)\}.$$

If $\mathrm{Dis}(\xi, \eta) > 0$, we say ξ and η are disjoint.

Theorem 2. *With the notations above. We use intervals to represent real numbers and use boxes to represent real points in the computation, if for any* $\eta_i^j \in \mathrm{Zero}(T_i^i)$, $\eta_{i-1} \in \mathrm{Zero}(T_{i-1}^{i-1})$, $|\eta_i^j - \eta_{i-1}| < (\prod_{j=1}^{i-2} \frac{r_j}{R_{j+1}}) r_i$, $i = 2, \ldots, n; j = 1, 2$,

$$\mathrm{Dis}(\eta_i^1, \eta_i^2) > |\eta_{i-1}|, \tag{14}$$

then any two real zeros $\boldsymbol{\xi}^1 = (\xi_1^1, \ldots, \xi_n^1)$ *and* $\boldsymbol{\xi}^2 = (\xi_1^2, \ldots, \xi_n^2)$ *of* Σ_n *are disjoint.*

3 The main algorithm

Algorithm 3 *Isolate the real (or complex) roots of a 0-dim. triangular system.*
Input: *A zero-dimensional triangular system* Σ_n*, a precision* ϵ*.*
Output: *The solutions of the system in isolating interval representation.*

1. Isolate the real (or complex) roots of $f_1(x_1) = 0$ under the precision $\rho = \frac{\epsilon}{20}$. Let $T_1^1(X_1^1) = f_1(X_1^1)$.
2. For i from 2 to n,
   (a) Estimate r_{i-1} with method in Section 2.2.
   (b) Estimate R_i with method in Section 2.2.
   (c) Compute $T_i^i(X_i^i)$ with method in Section 2.1.
   (d) Isolate the real roots of $T_i^i(X_i^i) = 0$ with precision $\Pi_{j=1}^{i-1} \frac{r_j}{R_{j+1}} \frac{\epsilon}{20}$ ($\Pi_{j=1}^{n-1} \frac{r_j}{R_{j+1}} \frac{\epsilon}{2}$ if $i = n$). Compute the multiplicities of the real roots if needed when $i = n$.
   (e) If (14) is not satisfied, then refine the real (or complex) roots of $T_{i-1}^{i-1}(X_{i-1}^{i-1}) = 0$ until (14) is satisfied.
   (f) Recover the real zeros of Σ_i from $T_i^i(X_i^i)$ and Σ_{i-1} by Theorem 1.
3. Get the algebraic solutions of Σ_n: $\{\{T_1^1(X_1^1), \ldots, T_n^n(X_n^n)\}, \{r_1, \ldots, r_{n-1}\}, \{R_2, \ldots, R_n\}\}$
   Or numeric solutions and their corresponding multiplicities.

Example 4 *Consider the system* $\{x^2 - 6, 5\,x^2 + 10\,xy + 6\,y^2 - 5, x^2 + 2\,xy + 2\,y^2 + 4\,yz + 5\,z^2 - 1\}$. *We derive a symbolic representation of the roots, as well as a floating point approximation up to precision* $\frac{1}{10^3}$. *We isolate the roots of* $f_1 = 0$ *using precision* $\frac{1}{2\cdot 10^4}$ *and we derive the zero set:*

$$H = \{\xi_1^1 = -2.449490070, \xi_1^2 = 2.449490070\}.$$

Let $r_1 = 2$. *Consider* $\xi_1 \approx -2.449490070 \in [-2.45, -2.44]$. *We can use* $-2.45, -2.44$ *to construct* $f^u(y), f^d(y)$ *for* $f_2(\xi_1, y)$. *We compute a root bound for* $f^u(y), f^d(y)$. *For both it is* ≤ 6. *Similarly, we compute a root bound for the other root in* H. *we notice that all the root bounds are less than 6. We have computed* $r_2 = 2$, *so we set* $R_2 = 6 + 2 = 2^3$. *By considering a coordinate system transformation, we derive a system* Σ_2' *as follows*

$$\{{X_2^2}^2 - \frac{1}{2}\,X_2^2\,X_1^2 + \frac{1}{16}\,{X_1^2}^2 - 6, 5\,{X_2^2}^2 + \frac{15}{2}\,X_2^2\,X_1^2 + \frac{61}{16}\,{X_1^2}^2 - 5\}$$

Hence we can compute $T_2^2 = 36\,{X_2^2}^4 - \frac{1083}{4}\,{X_2^2}^2 + \frac{130321}{256}$. *Solve* $T_2^2(X_2^2) = 0$ *under the precision* $\frac{1}{8\cdot 10^4}$, *we have its real roots and multiplicities (the number in each bracket is the multiplicity of the root in the system):* $G = \{\eta_2^1 = -1.939178944\,[2], \eta_2^2 = 1.939178944\,[2]\}$.

For each root η_2 *in* G, *if it satisfies* $|\eta_2 - \xi_1| < r_1 = 2$, *then it corresponds to* ξ_1, *where* ξ_1 *is a root in* H. *And the multiplicity of* (ξ_1, η_2) *in the given system is the corresponding multiplicity of* η_2 *in* $T_2^2 = 0$. *In this way, we can get the approximating roots of the subsystem* Σ_2:

$$\{[-2.449490070\,[1], 2.041244504\,[2]], [2.449490070\,[1], -2.041244504\,[2]]\}$$

With the method of Section 2.2, we estimate $r_3 = 2$, *and we derive that 3 is a bound for the* z *coordinate. Let* $R_3 = 2 + 2 = 4$ *and* $r_2 = 2$ *and consider a coordinate system transformation as mentioned above. By computing the resultant, we can get*

$$T_3^3 = 810000\,x^8 - 13500000\,x^6 + 84375000\,x^4 - 234375000\,x^2 + 244140625$$

Then, we get the solution of the given triangular system as follows.

$$\{\{{X_1^1}^2 - 6, 36\,{X_2^2}^4 - \tfrac{1083}{4}\,{X_2^2}^2 + \tfrac{130321}{256},$$
$$810000\,x^8 - 13500000\,x^6 + 84375000\,x^4 - 234375000\,x^2 + 244140625\}, \{2, 2\}, \{8, 4\}\}$$

We solve T_3^3 *using precision* $\frac{1}{16\cdot 10^4}$, *and derive its roots and multiplicities:*

$$J = \{\eta_3^1 = -2.041241452\,[4], \eta_3^2 = 2.041241452\,[4]\}.$$

For each root η_3 *in* J, *if it satisfies* $|\eta_3 - \eta_2| < \frac{r_1}{R_2} r_2 = \frac{1}{2}$, *then it corresponds to the same* (ξ_1, ξ_2) *with* η_2, *where* (ξ_1, ξ_2) *is a root in* Σ_2. *And the multiplicity of* (ξ_1, ξ_2, η_3) *in the given system is the corresponding multiplicity of* η_3. *In this way, we can get the approximating roots of the system:*

$$\{[-2.449490070\,[1], 2.041244504\,[2], -0.816497800\,[4]],$$
$$[2.449490070\,[1], -2.041244504\,[2], 0.816497800\,[4]]\}$$

Using Lemma 8, the precision of roots is $4(\frac{1}{8\cdot 10^4} + \frac{1}{16\cdot 10^4}) < \frac{1}{10^3}$.

4 Complexity Analysis

In what follows $\mathcal{O}_B$ means bit complexity and the $\tilde{\mathcal{O}}_B$-notation means that we are ignoring logarithmic factors. For a polynomial $f \in \mathbb{Z}[X]$, $\deg(f)$ denotes its degree. By $\mathcal{L}(f)$ we denote an upper bound on the bit size of the coefficients of f (including a bit for the sign), $\tilde{\mathcal{O}}$ indicates that we omit logarithmic factors. For $a \in \mathbb{Q}$, $\mathcal{L}(a)$ is the maximum bit size of the numerator and the denominator.

Lemma 9. *[22]For a polynomial f of degree d with integer coefficients of modulus less than 2^τ , we can isolate the real roots of f in $\tilde{\mathcal{O}}_B(d^3\tau)$.*

Lemma 10. *[2, 24] Let $f(x)$ be a polynomial in $\mathbb{Z}[x]$ and $\deg_x(f) \leq d$, $\mathcal{L}(f) \leq \tau$. Then the separation bound of f is $\mathrm{sep}(f) \geq d^{-\frac{d+2}{2}}(d+1)^{\frac{1-d}{2}}2^{\tau(1-d)}$, thus $\log(\mathrm{sep}(f)) = \tilde{\mathcal{O}}(d\tau)$. The latter provides a bound on the bit size of the endpoints of the isolating intervals.*

Lemma 11. *[11]Let $f, g \in (Z[y_1, \ldots, y_k])[x]$ with $\deg_x(f) = p \geq q = \deg_x(g)$, $\deg_{y_i}(f) \leq p$ and $\deg_{y_i}(g) \leq q$, $\mathcal{L}(f) = \tau \geq \sigma = \mathcal{L}(g)$. We can compute $\mathrm{Res}(f, g)$ w.r.t. x in $\tilde{\mathcal{O}}_B(q(p+q)^{k+1}p^k\tau)$. And $\deg_{y_i}(\mathrm{Res}_x(f, g)) \leq 2pq$, and the bit size of resultant is $\tilde{\mathcal{O}}(p\sigma + q\tau)$.*

Lemma 12. *Let $\Sigma_{k+1} = \{f_1(x_1), f_2(x_1, x_2), \ldots, f_{k+1}(x_1, x_2, \ldots, x_{k+1})\} \in \mathbb{Z}[x_1, \ldots, x_{k+1}]$. Assume $\deg_{x_j}(f_i(x_1, x_2, \ldots, x_i)) \leq d$, $\mathcal{L}(f_i(x_1, \ldots, x_i)) \leq \tau$, $1 \leq j \leq i$, $1 \leq i \leq k+1$. For any real numbers $\{\xi_1, \ldots, \xi_k\} \in Zero(\Sigma_k)$ represented by intervals $[a_1, b_1], \ldots, [a_k, b_k]$, assume $\mathcal{L}(\xi_i) \leq \sigma_i$. Then we compute R_{k+1} in $\tilde{\mathcal{O}}_B(kd^{\frac{(k+2)^2}{2}})$ and $\mathcal{L}(R_{k+1}) \leq \max\{\tilde{\mathcal{O}}(\sum_{i=1}^k \sigma_i d + \tau), \tilde{\mathcal{O}}(d^k\tau)\}$.*

Let $\mathbf{x}_i$ be the list $x_1, \ldots, x_i$. For $1 \leq i \leq n$, $1 \leq l \leq i-1$, let

$$\begin{aligned} \varphi_l^i : \mathbb{Z}[\mathbf{x}_{i-l+1}] &\to \mathbb{Q}[\mathbf{x}_{i-l-1}, X_{l+1}^i, X_l^i] \\ f(\mathbf{x}_{i-l+1}) &\mapsto f(\mathbf{x}_{i-l-1}, X_{l+1}^i - \tfrac{r_l}{R_{l+1}} X_l^i, X_l^i) \end{aligned} \tag{15}$$

Theorem 5. *Let $\Sigma_n = \{f_1(x_1), f_2(x_1, x_2), \ldots, f_n(x_1, x_2, \ldots, x_n)\} \in \mathbb{Z}[x_1, x_2, \ldots, x_n]$. Assume $\deg_{x_j}(f_i(x_1, x_2, \ldots, x_i)) \leq d$, $\mathcal{L}(f_i(x_1, x_2, \ldots, x_i)) \leq \tau$, where $1 \leq j \leq i$, and $1 \leq i \leq n$. Using Alg. 3 to isolate the real roots of Σ_n, we deduce:*

- *$\mathcal{L}(\xi_i) = \tilde{\mathcal{O}}(d^{i^2+2i-2}\tau)$, $1 \leq i \leq n$,*
- *$\mathcal{L}(r_i) = \tilde{\mathcal{O}}(d^{i^2+2i-2}\tau)$, and $\mathcal{L}(R_{i+1}) = \tilde{\mathcal{O}}(d^{i^2+2i-1}\tau)$, where $1 \leq i \leq n-1$,*
- *$\deg_{X_l^i} T_l^i(\mathbf{x}_{i-l}, X_l^i) \leq 3^{l-1}d^l$, $\deg_{x_j} T_l^i(\mathbf{x}_{i-l}, X_l^i) \leq 2^{l-1}d^l$, $\mathcal{L}(T_l^i) = \tilde{\mathcal{O}}(d^{i^2+l-2}\tau)$, where $2 \leq i-l$, $2 \leq l \leq i$, $2 \leq i \leq n$,*
- *We compute T_l^i in $\tilde{\mathcal{O}}_B(d^{i^2+(2l-2)i-2l^2+6l-5}\tau)$, and $\{\xi_i\}$ in $\tilde{\mathcal{O}}_B(d^{i^2+4i-2}\tau)$, where $2 \leq l \leq i$, $2 \leq i \leq n$.*

Theorem 6. *The complexity of Alg. 3 is $\tilde{\mathcal{O}}_B(N^{n^2})$, where $N = \max\{d, \tau\}$.*

5 Experiments

In this section, we illustrate the function of our algorithm by some examples. The timings are collected on a computer running Maple 15 with 2.29GHz CPU, 2G memory and Windows XP by using the time command in Maple.

We compare our method with Discover, Isolate, EVB and Vincent-Collins-Akritas algorithm. Discoverer is a tool for solving problems about polynomial equations and inequalities [23]. Isolate is a tool to solve general equation systems based on Realsolving C library by Rouillier. EVB is developed by Cheng et al in [9]. Vincent-Collins-Akritas algorithm which isolates real roots for univariate polynomials uses techniques which are very close to the ones used by Rioboo in [20]. Sqf is the method in [9] for zero-dimensional triangular system without multiple roots. All the required precision are 0.001.

In Table 1, we compare different methods by computing some zero-dimensional triangular polynomial systems without multiple roots. All the tested systems have the form $(f_1, f_2, \ldots, f_n)$. And $\deg(f_i) = k$ are the degrees of the polynomials. We take average timings for different degrees (each degree with several random examples).

In Table 2, we take polynomials with three variables. They are surfaces, denoted as f, in $\mathbb{R}^3$. We compute the resultant of f and $\frac{\partial f}{\partial z}$ with respect to z. Denote its squarefree part as g. Then we compute the resultant of g and $\frac{\partial g}{\partial y}$ with respect to y and denote the squarefree part as h. Thus we get a triangular polynomial system $\{h, g, f\}$. When computing the topology of real algebraic surfaces, one usually needs to solve this kind of triangular system. It is usually zero-dimensional. This kind of system always have multiple roots. We test this kind of zero-dimensional triangular systems for the methods which can deal with multiple roots directly. They are Isolate, EVB and LGP.

From the data, we can find that LGP works well for system with multiple roots comparing to the existing direct method. For the systems without multiple roots, Sqf is the most efficient method. LGP works well for system with fewer roots. For the systems with higher degrees or more variables, that is, systems with more roots, LGP will slow down comparing to other methods. The reason is that $\prod_{j=1}^{i-1} \frac{r_j}{R_{j+1}}$ becomes small, thus the resultant computations take much more time.

Acknowledgment. Partially supported by the EXACTA grant of the National Science Foundation of China (NSFC 60911130369) and the French National Re- search Agency (ANR-09-BLAN-0371-01).

References

1. M.-E. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Multiplicities and idempotents for zero dimensional systems. *In Algorithms in algebraic Geometry and Applications, Vol. 143 of Progress in Mathematics*, pages 1–20. Birkhäuser, 1996.
2. S. Basu, R. Pollack, and M. F. Roy. Algorithms in Real Algebraic Geometry, *Volume 10 of Algorithms and Computation in Mathematics*, Springer-Verlag, 2003.
3. E. Becker and T. Wörmann. Radical computations of zero-dimensional ideals and real root counting. *Math. and Computers in Simulation*, 42(4-6):561–569, 1996.

4. E. Berberich, M. Kerber, and M. Sagraloff. Exact Geometric-Topological Analysis of Algebraic Surfaces. In M. Teillaud, editor, Proc. of the 24th ACM Symp. on Computational Geometry (SoCG), pages 164–173. ACM press, 2008.
5. F. Boulier, C. Chen, F.Lemaire and M. Moreno Maza. Real Root Isolation of Regular Chains. ASCM 2009: 1-15, 2009.
6. J. S. Cheng, X. S. Gao, and J. Li. Root isolation for bivariate polynomial systems with local generic position method. ISSAC 2009: 103–110, 2009.
7. J. S. Cheng, X. S. Gao, and L. Guo. Root isolation of zero-dimensional polynomial systems with linear univariate representation. *J. of Symbolic Computation (2011)*.
8. J. S. Cheng, X. S. Gao, and M. Li. Determining the topology of real algebraic surfaces. In *Mathematics of Surfaces XI*, pp. 121–146, LNCS3604, Springer, 2005.
9. J. S. Cheng, X. S. Gao, and C. K. Yap. Complete Numerical Isolation of Real Roots in 0-dimensional Triangular Systems, *JSC*, 44(7): 768–785 (2009).
10. G. E. Collins, J. R. Johnson, and W. Krandick. Interval arithmetic in cylindrical algebraic decomposition. *Journal of Symbolic Computation*, 34:145–157, 2002.
11. D. I. Diochnos, I. Z. Emiris, and E. P. Tsigaridas. On the asymptotic and practical complexity of solving bivariate systems over the reals, *J. of Symbolic Computation*, 44(7):818–835,2009.
12. A. Eigenwillig, L. Kettner, W. Krandick, K. Mehlhorn, S. Schmitt, and N. Wolpert. A Descartes algorithm for polynomials with bit stream coefficients. in *CASC 2005*: 138–49, LNCS 3718, Springer, 2005.
13. I. Z. Emiris, B. Mourrain, and E. P. Tsigaridas. Real Algebraic Numbers: Complexity Analysis and Experimentation, *LNCS 5045*, pages 57–82, 2008.
14. I. Z. Emiris, B. Mourrain, and E. P. Tsigaridas. The DMM bound: Multivariate (aggregate) separation bounds. ISSAC 2010: 243–250, ACM, Germany, July 2010.
15. W. Fulton. Introduction to intersection theory in algebraic geometry, *Volume 54 of CBMS Regional Conference Series in Mathematics.* Published for the Conference Board of the Mathematical Sciences, Washington, DC, 1984
16. X. S. Gao and S. C. Chou. On the theory of resolvents and its applications, *Mathematics and Systems Science*, 1997.
17. H. Hong. An Efficient Method for Analyzing the Topology of Plane Real Algebraic Curves. *Mathematics and Computers in Simulation*, 42:571–582, 1996.
18. H. Hong and V. Stahl. Safe start region by fixed points and tightening. *Computing*, 53(3-4):323–335, 1994.
19. Z. Lu, B. He, Y. Luo and L. Pan. An Algorithm of Real Root Isolation for Polynomial Systems. SNC 2005.
20. R. Rioboo, Computation of the real closure of an ordered field, ISSAC 1992, Academic Press, San Francisco.
21. F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *AAECC*, 9:433–461, 1999.
22. M. Sagraloff. When Newton meets Descartes: A Simple and Fast Algorithm to Isolate the Real Roots of a Polynomial,CoRR abs/1109.6279: (2011)
23. B. Xia and T. Zhang. Real Solution Isolation Using Interval Arithmetic, *Computers and Mathematics with Applications*, 52:853–860, 2006.
24. C. Yap. *Fundamental Problems of Algorithmic Algebra*, Oxford University Press, New York, 2000.
25. C. Yap, M. Sagraloff. A simple but exact and efficient algorithm for complex root isolation. ISSAC 2011: 353–360.
26. Z.H. Zhang, T. Fang and B.C. Xia, Real solution isolation with multiplicity of 0-dimensional triangular systems, *Science China: Information Sciences*, 54(1): 60–69, 2011.

A Proofs

Table 1. Timing of Real Root Isolation of System without Multiple Roots (Seconds)

Degree	Vars	LGP	Dis	Iso	VCA	Sleeve	Sqf
2-11	2	0.155	0.325	0.254	1.071	0.887	0.024
12-20	2	4.224	3.242	23.106	7.915	39.438	0.076
2-4	3	0.113	0.336	0.202	1.774	0.152	0.045
5-7	3	9.063	3.118	45.771	11.178	79.953	0.110
2-3	4	0.175	0.498	0.715	2.115	0.199	0.024
4	4	10.008	2.727	70.350	13.250	24.041	0.121

Table 2. Timing of Real Root Isolate of surfaces(Seconds)

Degree	LGP	Iso	EVB
2	0.205	0.225	0.092
3	1.288	16.681	3.589
4	16.180	200.594	2337.999

Proof (of Lemma 1). Note that we derive the system
$\Theta_2 := \{f_i(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1}), f_{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1}, X_1^{i+1})\}$
from the system
$\Theta_1 := \{f_i(\xi_1, \ldots, \xi_{i-1}, x_i), f_{i+1}(\xi_1, \ldots, \xi_{i-1}, x_i, x_{i+1})\}$
by coordinate system transformation. So there exists a one to one correspondence between their zeros, including the multiplicities of the zeros by the properties of LGP method. And the coordinate system transformation ensures that for any zero (ξ_i, ξ_{i+1}), when projected to x_i-axis by LGP method, the zero is in the fixed neighborhood of ξ_i (centered at ξ_i bounded by $r_i/2$). This ensures that all the zeros of Θ_2, when projected to x_i-axis, do not overlap, which means any root of $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$ corresponds to one zero of Θ_2. So there exists a one to one correspondence between roots of $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$ and the zeros of Θ_1. It is not difficult to find that the degree of the polynomial $f_i(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1} - \frac{r_i}{R_{i+1}} X_1^{i+1})$ w.r.t. X_1^{i+1} is equal to its total degree. And $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1})$ is the resultant of the two polynomials in Θ_2 w.r.t. X_1^{i+1}. Based on the theory in Section 1.6 in [15], we can conclude that the multiplicities of the roots in $T_2^{i+1}(\xi_1, \ldots, \xi_{i-1}, X_2^{i+1}) = 0$ equals the multiplicities of the corresponding zeros of Θ_2, and then Θ_1. So we derive that the lemma is true.

Proof (of Lemma5). According to Lemma 4, we know

$$\begin{aligned}
\xi_i &= \tfrac{R_i}{r_{i-1}}(\zeta_2 - \xi_{i-1}) \\
&= \tfrac{R_i}{r_{i-1}}(\tfrac{R_{i-1}}{r_{i-2}}(\zeta_3 - \xi_{i-2}) - \xi_{i-1}) \\
&\quad \ldots \\
&= (\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})\eta_i - \sum_{k=1}^{i-1}[(\prod_{j=k}^{i-1} \tfrac{R_{j+1}}{r_j})\xi_k].
\end{aligned}$$

Note that here $\zeta_i = \eta_i$. Similarly, we have

$$\begin{aligned}
\xi_{i+1} &= (\textstyle\prod_{j=1}^{i} \tfrac{R_{j+1}}{r_j})\eta_{i+1} - \sum_{k=1}^{i}[(\prod_{j=k}^{i} \tfrac{R_{j+1}}{r_j})\xi_k] \\
&= (\textstyle\prod_{j=1}^{i} \tfrac{R_{j+1}}{r_j})\eta_{i+1} - \sum_{k=1}^{i-1}[(\prod_{j=k}^{i} \tfrac{R_{j+1}}{r_j})\xi_k] \\
&\quad - \tfrac{R_{i+1}}{r_i}\xi_i \\
&= (\textstyle\prod_{j=1}^{i} \tfrac{R_{j+1}}{r_j})\eta_{i+1} - \tfrac{R_{i+1}}{r_i}\sum_{k=1}^{i-1}[(\prod_{j=k}^{i-1} \tfrac{R_{j+1}}{r_j})\xi_k] \\
&\quad - \tfrac{R_{i+1}}{r_i}\xi_i \\
&= (\textstyle\prod_{j=1}^{i} \tfrac{R_{j+1}}{r_j})\eta_{i+1} - \tfrac{R_{i+1}}{r_i}\sum_{k=1}^{i-1}[(\prod_{j=k}^{i-1} \tfrac{R_{j+1}}{r_j})\xi_k] \\
&\quad - \tfrac{R_{i+1}}{r_i}((\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})\eta_i - \sum_{k=1}^{i-1}[(\prod_{j=k}^{i-1} \tfrac{R_{j+1}}{r_j})\xi_k]) \\
&= (\textstyle\prod_{j=1}^{i} \tfrac{R_{j+1}}{r_j})(\eta_{i+1} - \eta_i).
\end{aligned}$$

Then we have

$$\begin{aligned}
|\eta_{i+1} - \eta_i| &= \textstyle\prod_{j=1}^{i} \tfrac{r_j}{R_{j+1}}|\xi_{i+1}| \\
&< \textstyle\prod_{j=1}^{i} \tfrac{r_j}{R_{j+1}}R_{i+1} \\
&= (\textstyle\prod_{j=1}^{i-1} \tfrac{r_j}{R_{j+1}})r_i.
\end{aligned}$$

The lemma has been proved.

Proof (of Thm. 2). We need only to consider the case η_i^1, η_i^2 are in the neighborhood of η_{i-1}. Otherwise, they are obviously disjoint. According to (8), for any $i = 2, \ldots, n$,

$$\begin{aligned}
\xi_i^1 &= (\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})(\eta_i^1 - \eta_{i-1}), \\
\xi_i^2 &= (\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})(\eta_i^2 - \eta_{i-1}).
\end{aligned}$$

If (14) is satisfied,

$$\begin{aligned}
\mathrm{Dis}(\xi_i^1, \xi_i^2) &= (\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})\mathrm{Dis}(\eta_i^1 - \eta_{i-1}, \eta_i^2 - \eta_{i-1}) \\
&\geq (\textstyle\prod_{j=1}^{i-1} \tfrac{R_{j+1}}{r_j})(\mathrm{Dis}(\eta_i^1, \eta_i^2) - |\eta_{i-1}|) \\
&> 0.
\end{aligned}$$

So, $\mathrm{Dis}(\boldsymbol{\xi}^1, \boldsymbol{\xi}^2) > 0$.

Proof (of Lemma 12). According to section 2.2, we may get R_{k+1} using two different methods in two different cases.

In the first case, we compute R_{k+1} by (2). R_{k+1} is the maximal one in the root bounds of $f_{k+1}(\xi_1, \ldots, \xi_k, x_{k+1})$ for all $\{\xi_1, \ldots, \xi_k\} \in \mathrm{Zero}(\Sigma_k)$. The root bound of $f_{k+1}(\xi_1, \ldots, \xi_k, x_{k+1})$ is the larger between the root bound of

$f_{k+1}^u(x_{k+1})$ and $f_{k+1}^d(x_{k+1})$. Note that $f_{k+1}^u(x_{k+1}) = f_{k+1}^+(b_1, \ldots, b_k, x_{k+1}) - f_{k+1}^-(a_1, \ldots, a_k, x_{k+1}) \in \mathbb{Q}[x_{k+1}]$ is a polynomial with degree less than d and bit size bounded by $\sum_{i=1}^k \sigma_i d + \tau$. By lemma 7, if $\mathcal{L}(f) \leq \tau$, then $\mathcal{L}(R) \leq \tau$ where R is the root bound of $f(x)$. Then the bit size of the root bound of $f_{k+1}^u(x)$ is bounded by $\sum_{i=1}^k \sigma_i d + \tau$. Similarly, the bit size of the root bound of $f_{k+1}^d(x_{k+1})$ is bounded by $\sum_{i=1}^k \sigma_i d + \tau$.

In the second case, we compute R_{k+1} by computing the root bound of $S(x_{k+1})$ defined in (9). First we prove that we can compute $S(x_{k+1})$ in $\tilde{\mathcal{O}}_B(kd^{\frac{(k+2)^2}{2}})$ and $\mathcal{L}(S(x_{k+1})) \leq \tilde{\mathcal{O}}(d^k\tau)$. Define

$$\begin{aligned}
S_2 &= \mathrm{Res}_{x_k}(f_{k+1}, f_k), \\
&\ldots, \\
S_{i+1} &= \mathrm{Res}_{x_{k-i+1}}(S_i, f_{k-i+1}), \\
&\ldots, \\
S_{k+1} &= \mathrm{Res}_{x_1}(S_k, f_1).
\end{aligned}$$

Then $S = S_{k+1}$. We prove following conclusions by inductive method:

(1) We compute $S_{i+1} = \mathrm{Res}_{x_{k-i+1}}(S_i, f_{k-i+1})$ in

$$\tilde{\mathcal{O}}_B(d^{2i(k-i+2)}\tau) \leq \tilde{\mathcal{O}}_B(d^{\frac{(k+2)^2}{2}}\tau);$$

(2)$\mathcal{L}(S_i) \leq \tilde{\mathcal{O}}(d^i\tau)$.

For $i = 1$, by lemma 11, we compute S_2 in $\tilde{\mathcal{O}}_B(d(2d)^{k+1}d^k\tau) = \tilde{\mathcal{O}}_B(d^{2(k+1)}\tau) \leq \tilde{\mathcal{O}}_B(d^{\frac{(k+2)^2}{2}}\tau)$ and $\mathcal{L}(S_2) \leq \tilde{\mathcal{O}}(2d\tau) = \tilde{\mathcal{O}}(d\tau)$, $\deg_{x_j}(S_2) \leq 2d^2$.

Assume we prove above conclusions for $1, 2, \ldots, i-1$.

For i, $S_{i+1} = \mathrm{Res}_{x_{k-i+1}}(S_i, f_{k-i+1})$. $\mathcal{L}(S_i) \leq 2^{i-1}d^{i-1}\tau$ and $\deg_{x_j}(S_i) \leq 2^{i-1}d^i$. According to lemma 11, we compute S_{i+1} in $\tilde{\mathcal{O}}_B(d(2^{i-1}d^i + d)^{k-i+2}(2^{i-1}d^i)^{k-i+1}2^{i-1}d^{i-1}\tau) = \tilde{\mathcal{O}}_B(d^{2i(k-i+2)}\tau) \leq \tilde{\mathcal{O}}_B(d^{\frac{(k+2)^2}{2}}\tau$ and $\mathcal{L}(S_{i+1}) \leq \tilde{\mathcal{O}}(2^{i-1}d^{i-1}\tau d + 2^{i-1}d^i\tau) = \tilde{\mathcal{O}}(d^i\tau)$.

Hence, we have proved above two conclusions.

According to above discussion, we can compute $S(x_{k+1})$ in

$$\tilde{\mathcal{O}}_B(kd^{\frac{(k+2)^2}{2}}\tau)$$

and $\mathcal{L}(S_{k+1}) \leq \tilde{\mathcal{O}}(d^k\tau)$. By lemma 7, the bit size of $\mathrm{rb}(S(x_{k+1}))$ is bounded by $\tilde{\mathcal{O}}(d^k\tau)$.

In conclusion, we proved this lemma.

Proof (of Thm. 5). For any $k = 1, 2, \ldots, n$, let $\mathcal{L}(r_k)$, $\mathcal{L}(R_{k+1})$, $\mathcal{L}(\xi_k)$ be bounded by $\rho_k, \tau_{k+1}, \sigma_k$ respectively. Furthermore, we can always assume ξ_k to be represented by an interval $[a_k, b_k]$ where a_k, b_k are fractions with denominators in the form of $2^{t_k}(t_k \leq \sigma_k)$ and numerator being 1, r_k to be in the form $\frac{1}{2^{p_k}}(p_k \leq \rho_k)$ and R_{k+1} to be in the form $2^{q_{k+1}}(q_{k+1} \leq \tau_{k+1})$. Then rational number $\frac{r_k}{R_{k+1}}$ are in the form $\frac{1}{2^{p_k+q_{k+1}}}$
$(p_k + q_{k+1} \leq \rho_k + \tau_{k+1})$.

We prove this theorem using inductive method.

For $i = 1$, we will compute $\{\xi_1\}, r_1, R_2$. According to Lemma 9 and Lemma 10, we isolate $\{\xi_1\}$ in $\tilde{\mathcal{O}}_B(d^3\tau)$, and $\mathcal{L}(b_1) \le \tilde{\mathcal{O}}(d\tau)$, $\mathcal{L}(r_1) \le \tilde{\mathcal{O}}(d\tau)$. According to Lemma 12, $\mathcal{L}(R_2) \le \max\{\tilde{\mathcal{O}}(d^2\tau + \tau), \tilde{\mathcal{O}}_B(d\tau)\} = \tilde{\mathcal{O}}(d^2\tau)$. Then (a)(b)(c)(f) is correct for $i = 1$.

For $i = 2$, we will compute T_2^2, $\{\xi_2\}, r_2, R_3$.

$T_2^2(X_2^2) = \mathrm{Res}_{X_1^2}((2^{\tau_2+\rho_1})^d\varphi_1^2(f_1), (2^{\tau_2+\rho_1})^d\varphi_1^2(f_2))$, where $(2^{\tau_2+\rho_1})^d\varphi_1^2(f_1)$ and $(2^{\tau_2+\rho_1})^d\varphi_1^2(f_2))$ are polynomials with integer coefficients. Furthermore, $(2^{\tau_2+\rho_1})^d\varphi_1^2(f_1)$ is degree less than d w.r.p.t X_1^2 and bit size less than $\tilde{\mathcal{O}}(2(\tau_2 + \rho_1)d + \tau) = \tilde{\mathcal{O}}(d\tau_2) = \tilde{\mathcal{O}}(d^3\tau)$. Similarly, $(2^{\tau_2+\rho_1})^d\varphi_1^2(f_2)) \in \mathbb{Z}[X_1^2, X_2^2]$ is degree less than $2d$ w.r.t. X_1^2 and bit size less than $\tilde{\mathcal{O}}(d^3\tau)$. According to Lemma 11, we compute $T_2^2(X_2^2)$ in $\tilde{\mathcal{O}}_B(d(2d+d)^{1+1}(2d)d^3\tau) = \tilde{\mathcal{O}}_B(18d^7\tau)$, and $\deg_{X_2^2}(T_2^2) \le 3d^2$, bit size of T_2^2 is bounded by $\tilde{\mathcal{O}}(3d^4\tau)$, $T_2^2 \in \mathbb{Z}[X_2^2]$. By Lemma 9, we isolate $\{\eta_2\}$, the real roots of T_2^2 in $\tilde{\mathcal{O}}_B(3^4d^{10}\tau) = \tilde{\mathcal{O}}_B(d^{10}\tau)$, and the end points of the isolate intervals of them have bit size bounded by $\tilde{\mathcal{O}}(3^2d^6\tau) = \tilde{\mathcal{O}}(d^6\tau)$. Then, we get back $\{\xi_2\}$ by $\xi_2 = \frac{R_2}{r_1}(\eta_2 - \xi_1)$, the bit size of ξ_2 are bounded by $\tilde{\mathcal{O}}(\tau_2 + \rho_1 + \mathcal{L}(\eta_2)) = \tilde{\mathcal{O}}((d^6 + d^3 + d)\tau) = \tilde{\mathcal{O}}(d^6\tau)$. So $\mathcal{L}(r_2) \le \tilde{\mathcal{O}}(d^6\tau)$. According to Lemma 12, $\mathcal{L}(R_3) \le \max\{\tilde{\mathcal{O}}(d^7\tau + d^2\tau + \tau), \tilde{\mathcal{O}}_B(d^2\tau)\} = \tilde{\mathcal{O}}(d^7\tau)$. Obviously, (a)-(f) have been proved for $i = 2$. Assume conclusions have been proved for $1, 2, \ldots, i-1$.

For i, we will compute $T_l^i, l = 2, 3, \ldots, i$, $\{\xi_i\}, r_i, R_{i+1}$.

We will induce l in the following discussion.

For $l = 2$, we will compute T_2^i. $T_2^i(X_2^i) = \mathrm{Res}_{X_1^i}((2^{\tau_i+\rho_{i-1}})^d \varphi_{i-1}^i(f_{i-1}), (2^{\tau_i+\rho_{i-1}})^d\varphi_{i-1}^i(f_i))$, where $(2^{\tau_i+\rho_{i-1}})^d\varphi_{i-1}^i(f_{i-1})$ and $(2^{\tau_i+\rho_{i-1}})^d\varphi_{i-1}^i(f_i))$ are polynomials with integer coefficients. Furthermore, $(2^{\tau_i+\rho_{i-1}})^d\varphi_{i-1}^i(f_{i-1})$ is degree less than d w.r.p.t X_1^i and bit size less than $\tilde{\mathcal{O}}(2(\tau_i + \rho_{i-1})d + \tau) = \tilde{\mathcal{O}}(d\tau_i)$. Similarly, $(2^{\tau_i+\rho_{i-1}})^d\varphi_{i-1}^i(f_i) \in \mathbb{Z}[\mathbf{x}_{i-2}, X_i^i, X_{i-1}^i]$ is degree less than $2d$ w.r.p.t X_{i-1}^i and bit size less than $\tilde{\mathcal{O}}(d\tau_i)$. According to Lemma 11, we compute $T_2^i(\mathbf{x}_{i-2}, X_2^i)$ in $\tilde{\mathcal{O}}_B(d(2d+d)^i(2d)^{i-1}d\tau_i) = \tilde{\mathcal{O}}_B(2^{i-1}3^id^{2i+1}d^{i^2-2}\tau) = \tilde{\mathcal{O}}_B(d^{i^2+2i-1}\tau)$, and $\deg_{X_2^i}(T_2^i) \le 3d^2$, $\deg_{x_j}(T_2^i) \le 2d^2, j = 1, \ldots, i-2$, bit size of T_2^i is bounded by $\tilde{\mathcal{O}}(d^2\tau_i) = \tilde{\mathcal{O}}(d^{i^2}\tau)$, $T_2^i \in \mathbb{Z}[\mathbf{x}_{i-2}, X_2^i]$.

Assume (d)(e) have been proved for $2, 3, \ldots, l-1$.

For l, we compute T_l^i. Similarly to $l = 2$, $T_l^i(X_l^i) = \mathrm{Res}_{X_{l-1}^i}$ $((2^{\tau_{i-l+2}+\rho_{i-l+1}})^d\varphi_{i-l+1}^i(f_{i-l+1}), (2^{\tau_{i-l+2}+\rho_{i-l+1}})^{3^{l-2}d^{l-1}}$ $\varphi_{i-l+1}^i(T_{l-1}^i))$, where $(2^{\tau_{i-l+2}+\rho_{i-l+1}})^d\varphi_{i-l+1}^i(f_{i-l+1})$ and $(2^{\tau_{i-l+2}+\rho_{i-l+1}})^{3^{l+2}d^{l-1}}\varphi_{i-l+1}^i(T_{l-1}^i))$ are polynomials with integer coefficients. Furthermore, $(2^{\tau_{i-l+2}+\rho_{i-l+1}})^d\varphi_{i-l+1}^i$ (f_{i-l+1}) is degree less than d w.r.p.t X_{l-1}^i, bit size less than $\tilde{\mathcal{O}}(2(\tau_{i-l+2} + \rho_{i-l+1})d + \tau) = \tilde{\mathcal{O}}(d\tau_{i-l+2})$. Similarly, $(2^{\tau_{i-l+2}+\rho_{i-l+1}})^{3^{l-2}d^{l-1}}\varphi_{i-l+1}^i(T_{l-1}^i)) \in \mathbb{Z}[\mathbf{x}_{i-l+2}, X_l^i, X_{l-1}^i]$ is degree less than $2\cdot 3^{l-2}d^{l-1}$ w.r.p.t X_{l-1}^i and bit size less than $\tilde{\mathcal{O}}(3^{l-2}d^{l-1}\tau_{i-l} + d^{l-1}\tau_i) = \tilde{\mathcal{O}}(d^{l-1}\tau_i)$. According to Lemma 11, we compute $T_l^i(X_l^i)$ in $\tilde{\mathcal{O}}_B(d(3^{l-2}d^{l-1} + d)^{i-l+2}$

$(3^{l-2}d^{l-1})^{i-l+1}d^{l-1}\tau_i) = \tilde{\mathcal{O}}_B(d^{2(l-1)i--2l^2+6l-3}\tau_i) = \tilde{\mathcal{O}}_B$ $(d^{i^2+2(l-1)i-2l^2+6l-5}\tau)$, and $\deg_{X_l^i}(T_l^i) \le 3^{l-1}d^l$, $\deg_{x_j}(T_l^i)$ $\le 2^{l-1}d^l, j = 1,\ldots,i-l$, bit size of T_l^i is bounded by $\tilde{\mathcal{O}}(d^l\tau_i) = \tilde{\mathcal{O}}(d^{i^2+l-2}\tau)$, $T_l^i \in \mathbb{Z}[\mathbf{x}_{i-l}, X_l^i]$. So (d)(e) have been proved for l.

Furthermore, we compute T_i^i in $\tilde{\mathcal{O}}_B(d^{i^2+4i-5}\tau)$, $\deg_{X_i^i}(T_i^i)$ $\le 3^{i-1}d^i$, bit size of T_i^i is bounded by $\tilde{\mathcal{O}}(d^{i^2+i-2}\tau)$. By Lemma 9, we isolate $\{\eta_i\}$, the real roots of T_i^i in $\tilde{\mathcal{O}}_B((3^{i-1}d^i)^3$ $d^{i^2+2i-3}\tau) = \tilde{\mathcal{O}}_B(d^{i^2+5i-3}\tau)$, and the end points of the isolate intervals of them have bit size bounded by $\tilde{\mathcal{O}}(3^{i-1}d^i$ $3d^{i^2+i-2}\tau) = \tilde{\mathcal{O}}(d^{i^2+2i-2}\tau)$. Then, we get back $\{\xi_i\}$ by $\xi_i = \frac{R_i}{r_{i-1}}(\eta_i - \xi_{i-1})$, the bit size of ξ_2 are bounded by $\tilde{\mathcal{O}}(\tau_i + \rho_{i-1} + \mathcal{L}(\eta_i)) = \tilde{\mathcal{O}}(d^{i^2-2}\tau + d^{i^2-3}\tau + d^{i^2+2i-2}\tau) = \tilde{\mathcal{O}}(d^{i^2+2i-2}\tau)$. So $\mathcal{L}(r_i) \le \tilde{\mathcal{O}}(d^{i^2+2i-2}\tau)$. According to Lemma 12, $\mathcal{L}(R_{i+1}) \le \max\{\tilde{\mathcal{O}}(d^{i^2+2i-1}\tau), \tilde{\mathcal{O}}_B(d^i\tau)\} = \tilde{\mathcal{O}}(d^{i^2+2i-1}\tau)$. Obviously, this theorem have been proved.

Proof (of Thm. 6). In Algorithm 3, we need compute R_i, T_l^i for $i = 2,\ldots,n; l = 2,\ldots,i$, and isolate the real roots of T_i^i for $i = 1,\ldots,n$. So the complexity of this algorithm is

$$\begin{aligned}&\textstyle\sum_{i=1}^{n-1}\tilde{\mathcal{O}}_B(id^{\frac{(i+2)^2}{2}}\tau)\\ &+\textstyle\sum_{i=2}^{n}\sum_{l=2}^{i}\tilde{\mathcal{O}}_B(d^{i^2+(2l-2)i-2l^2+6l-5}\tau)\\ &+\textstyle\sum_{i=1}^{n}\tilde{\mathcal{O}}_B(d^{i^2+4i-2}\tau) = \tilde{\mathcal{O}}_B(N^{n^2}).\end{aligned}$$